

---

# Learning Summary Statistic for Approximate Bayesian Computation via Deep Neural Network

Bai Jiang, Tung-yu Wu, Charles Zheng and Wing H. Wong

*Stanford University*

*Abstract:* Approximate Bayesian Computation (ABC) methods are used to approximate posterior distributions in models with unknown or computationally intractable likelihoods. Both the accuracy and computational efficiency of ABC depend on the choice of summary statistic, but outside of special cases where the optimal summary statistics are known, it is unclear which guiding principles can be used to construct effective summary statistics. In this paper we explore the possibility of automating the process of constructing summary statistics by training deep neural networks to predict the parameters from artificially generated data: the resulting summary statistics are approximately posterior means of the parameters. With minimal model-specific tuning, our method constructs summary statistics for the Ising model and the moving-average model, which match or exceed theoretically-motivated summary statistics in terms of the accuracies of the resulting posteriors.

*Key words and phrases:* Approximate Bayesian Computation, Summary Statistic, Deep Learning

## 1. Introduction

Bayesian inference is traditionally centered around the ability to compute or sample from the posterior distribution of the parameters, having conditioned on the observed data. Suppose data  $X$  is generated from a model  $\mathcal{M}$  with parameter  $\theta$ , the prior of which is denoted by  $\pi(\theta)$ . If the closed form of the likelihood function  $l(\theta) = p(X|\theta)$  is available, the posterior distribution of  $\theta$  given observed data  $x_{obs}$  can be computed via Bayes' rule

$$\pi(\theta|x_{obs}) = \frac{\pi(\theta)p(x_{obs}|\theta)}{p(x_{obs})}.$$

Alternatively, if the likelihood function can only be computed conditionally or up to a normalizing constant, one can still draw samples from the posterior by using stochastic simulation techniques such as Markov Chain Monte Carlo (MCMC) and rejection sampling (Asmussen and Glynn (2007)).

However, in many applications, the likelihood function  $l(\theta) = p(X|\theta)$  cannot be explicitly obtained, or is intractable to compute; this precludes the possibility of direct computation or MCMC sampling. In these cases, approximate inference can still be performed so long as 1) it is possible to draw  $\theta$  from the prior  $\pi(\theta)$ , and 2) it is possible to simulate  $X$  from the model  $\mathcal{M}$  given  $\theta$ , using the methods of Approximate Bayesian Computation (ABC) (See e.g. Beaumont, Zhang and Balding (2002); Lopes and Beaumont (2010); Beaumont (2010); Csilléry, Blum, Gaggiotti and François (2010); Marin, Pudlo, Robert and Ryder (2012); Toni, Welch, Strelkowa, Ipsen and Stumpf (2009); Sunnåker, Busetto, Numminen, Corander, Foll and Dessimoz (2013)).

While many variations of the core approach exist, the fundamental idea underlying ABC is quite simple: that one can use rejection sampling to obtain draws from the posterior distribution  $\pi(\theta|x_{obs})$  without computing any likelihoods. We draw parameter-data pairs  $(\theta', X')$  from the prior  $\pi(\theta)$  and the model  $\mathcal{M}$ , and accept only the  $\theta'$  such that  $X' = x_{obs}$ , which occurs with conditional probability  $p(x_{obs}|\theta')$  for any  $\theta'$ . Algorithm 1 describes the ABC method for discrete data (Tavaré, Balding, Griffiths and Donnelly (1997)), which yields an i.i.d. sample  $\{\theta^{(i)}, 1 \leq i \leq n\}$  of the exact posterior distribution  $\pi(\theta|X = x_{obs})$ .

---

**Algorithm 1** ABC rejection sampling 1

---

```

for  $i = 1, \dots, n$  do
  repeat
    Propose  $\theta' \sim \pi(\theta)$ 
    Draw  $X' \sim \mathcal{M}$  given  $\theta'$ 
  until  $X' = x_{obs}$  (acceptance criterion)
  Accept  $\theta'$  and let  $\theta^{(i)} = \theta'$ 
end for

```

---

The success of Algorithm 1 depends on acceptance rate of proposed parameter  $\theta'$ , i.e. the probability of the event  $X' = x_{obs}$  given  $\theta$ . For continuous  $x_{obs}$  and  $X'$ , the event  $X' = x_{obs}$  happens with probability 0, and hence Algorithm

1 is unable to produce any draws. As a remedy, one can relax the acceptance criterion  $X' = x_{obs}$  to be  $\|X' - x_{obs}\| < \epsilon$ , where  $\|\cdot\|$  is a norm and  $\epsilon$  is the tolerance threshold. The choice of  $\epsilon$  is crucial for balancing efficiency and approximation error, since with smaller  $\epsilon$  the approximation error decreases while the acceptance probability also decreases.

However, when the observation vectors are high-dimensional, the inefficiency of rejection sampling in high dimensions results in either extreme inaccuracy, or accuracy at the expense of an extremely time-consuming procedure. To circumvent the problem, one can introduce low-dimensional summary statistic  $S$  and further relax the acceptance criterion to be  $\|S(X') - S(x_{obs})\| < \epsilon$ . The use of summary statistics results in Algorithm 2, which was first proposed as the extension of Algorithm 1 in population genetics application (Fu and Li (1997); Weiss and von Haeseler (1998); Pritchard, Seielstad, Perez-Lezaun and Feldman (1999)).

---

**Algorithm 2** ABC rejection sampling 2

---

```

for  $i = 1, \dots, n$  do
  repeat
    Propose  $\theta' \sim \pi$ 
    Draw  $X' \sim \mathcal{M}$  with  $\theta'$ 
  until  $\|S(X') - S(x_{obs})\| < \epsilon$  (relaxed acceptance criterion)
  Accept  $\theta'$  and let  $\theta^{(i)} = \theta'$ 
end for

```

---

Instead of the exact posterior distribution, the resulting sample  $\{\theta^{(i)}, 1 \leq i \leq n\}$  obtained by Algorithm 2 follows an approximate posterior distribution

$$\pi(\theta | \|S(X') - S(x_{obs})\| < \epsilon) \approx \pi(\theta | S(X) = S(x_{obs})) \quad (1.1)$$

$$\approx \pi(\theta | X = x_{obs}). \quad (1.2)$$

The choice of the summary statistic is crucial for the approximation quality of ABC posterior distribution. A good summary statistic should offer a good trade-off between two approximation errors (Blum, Nunes, Prangle, Sisson and others (2013)). The approximation error (1.1) is introduced when one replaces “equal to” with “similar” in the first relaxation of the acceptance criterion. Under appropriate regularity conditions, it vanishes as  $\epsilon \rightarrow 0$ . The approximation error

(1.2) is introduced when one compares summary statistics  $S(X)$  and  $S(x_{obs})$  rather than the original data  $X$  and  $x_{obs}$ . In essence, this is just the information loss of mapping high-dimensional  $X$  to low-dimensional  $S(X)$ . A summary statistic  $S$  of higher dimension is in general more informative, hence reduces the approximation error (1.2). But at the same time, increasing the dimension of the summary statistic slows down the rate that the approximation error (1.1) vanishes in the limit of  $\epsilon \rightarrow 0$ . Ideally, we seek a statistic which is simultaneously low-dimensional and informative.

A sufficient statistic is an attractive option, since sufficiency, by definition, implies that the approximation error (1.2) is zero (Lehmann and Casella (1998)). However, the sufficient statistic has generally the same dimensionality as the sample size, except in special cases such as exponential families. And even when a low-dimensional sufficient statistic exists, it may be intractable to compute.

The main task of this article is to construct low-dimensional and informative summary statistics for ABC methods. Since our goal is to compare methods of constructing sufficient statistics (rather than present a complete methodology for ABC), the relatively simple Algorithm 2 suffices. In future work, we plan to use our approach for constructing summary statistics alongside more sophisticated variants of ABC methods, such as those which combine ABC with Markov chain Monte Carlo or sequential techniques (Marjoram, Molitor, Plagnol and Tavaré (2003); Sisson, Fan and Tanaka (2007)). Hereafter all ABC procedures we mentioned in this paper use Algorithm 2.

Existing methods for constructing summary statistics can be roughly classified into two classes (Blum, Nunes, Prangle, Sisson and others (2013)), both of which require a set of candidate summary statistics. The first class consists of approaches for *best subset selection*. Subsets of the candidate set of summary statistics are evaluated according to various information-based criteria, e.g. measure of sufficiency (Joyce and Marjoram (2008)), entropy (Nunes and Balding (2010)), AIC (Blum, Nunes, Prangle, Sisson and others (2013)) and BIC (Blum, Nunes, Prangle, Sisson and others (2013)), and the “best” subset is chosen to be the summary statistic. The second class is *projection* approach, which constructs summary statistics by linear or non-linear regression on the candidate set (Boulesteix and Strimmer (2007); Wegmann, Leuenberger and Excoffier

(2009); Blum and François (2010); Fearnhead and Prangle (2012)). Many of these methods require both expert knowledge and candidate summary statistics. An exception is Fearnhead and Prangle (2012)’s semi-automatic method, in which the authors take powers of individual data points as the initial candidate set and use linear regression to construct summary statistics.

Here we use deep neural networks (DNN) to automatically learn summary statistics for high-dimensional  $X$ . Blum and François (2010) has also considered the use of artificial neural networks for nonlinear projection; however, we are the first to consider the use of multilayer neural networks. The additional representational power offered by DNNs (compared to single-layer networks or other nonlinear function classes) enables our approach to be effective even without the need to specify an initial set of candidate statistics. In all of our experiments, we simply use the original data points as the input. Since we rely on the DNNs to automatically learn the appropriate nonlinear transformations which are necessary to construct useful summaries from the raw data, there is also no need to consider choosing a basis expansion of the original data points as in Fearnhead and Prangle (2012)’s semi-automatic method. Thus our choice of using DNNs allows to achieve a much higher degree of automation in constructing summary statistics than previous approaches.

Our procedure for constructing summary statistics is as follows:

- (i) Generate a training set  $\{\theta^{(i)}, X^{(i)}\}$  from prior  $\pi(\theta)$  and the model  $\mathcal{M}$ ,
- (ii) Train a DNN, with  $\{X^{(i)}\}$  as input and  $\{\theta^{(i)}\}$  as target,
- (iii) Use the estimator  $\hat{\theta}(X)$  as summary statistics in ABC procedure.

The DNNs in our approach compose non-linear transformations of data in the hidden layers, and fit linear regressions in the output layer with neurons in the top hidden layer as explanatory variables.

The idea is inspired by the notable successes of deep learning approach in several areas of machine learning, especially computer vision and natural language processing (Hinton and Salakhutdinov (2006); Hinton, Osindero and Teh (2006); Bengio, Courville and Vincent (2013); Schmidhuber (2015)). More and more practical and theoretical results show that deep architectures composed of simple learning modules in multiple layers can model high-level abstraction in

high-dimensional data. Thus it is expected that DNNs can effectively learn a good approximation to the posterior mean  $\hat{\theta}(X) \approx \mathbb{E}_{\pi}(\theta|X)$  under squared error loss, given a sufficiently large training set.

Our motivation for using an approximation of  $\mathbb{E}_{\pi}(\theta|X)$  as a summary statistic for ABC is inspired by results of Fearnhead and Prangle (2012) which demonstrate that the use of the posterior mean  $\mathbb{E}_{\pi}(\theta|X)$  optimizes certain criteria for first-order optimality. In a theoretical section, we extend their results and provide a simple proof in Theorem 1 based on conditioning, which is different to Fearnhead and Prangle (2012)’s proof via density-based calculation. In two simulated experiments, we construct summary statistics for Ising model and moving-average model of order 2. Ising model has sufficient statistic, which is the ideal summary statistic but a highly non-linear function in high-dimensional space. It’s a challenging task to construct a summary statistic akin to such a sufficient statistic due to high non-linearity and high-dimensionality. However, we see in our experiments that the DNN-based summary statistic approximates an increasing function of the sufficient statistic. In contrast, the semi-automatic summary statistic is unable to capture information about interactions, and hence fails to approximate the sufficient statistic. For moving-average model of order 2, the DNN-based summary statistic outperforms the auto-covariance statistic, and the semi-automatic construction. It is noteworthy that an automatically constructed summary statistic can outperform auto-covariance in MA(2) model, since the auto-covariance can be transformed to yield a consistent estimate of the parameters, and was widely used in the literature.

The rest of the article is organized as follows. In Section 2, we show how to construct summary statistics using deep neural networks. In Sections 3 and 4, we report the simulation studies on the Ising model and the moving average model, respectively. We describe in the supplementary materials the implementation details of training deep neural networks and other theoretical result, namely, how consistency can be obtained by using the posterior mean of a basis of functions of the parameters.

## 2. Methods

Throughout the paper, we denote by  $\mathcal{M}$  the model,  $X \in \mathbb{R}^p$  the data, and  $\theta \in \mathbb{R}^q$  the parameter. We assume it is possible to obtain a large number of

independent draws  $X \sim p(\cdot|\theta)$ . Denote by  $x_{obs}$  the observed data,  $\pi$  the prior of  $\theta$ ,  $S$  the summary statistic,  $\|\cdot\|$  the norm to measure  $S(X) - S(x_{obs})$ , and  $\epsilon$  the tolerance threshold. Let  $\pi_{ABC}^\epsilon(\theta) = \pi(\theta | \|S(X) - S(x_{obs})\| < \epsilon)$  denote the approximate posterior distribution obtained by Algorithm 2.

The main problem we address is how to construct a low-dimensional and informative summary statistic  $S$  for high dimensional  $X$ , which will enable accurate approximation of  $\pi_{ABC}^\epsilon$ . We are interested mainly in the regime where ABC is most effective: settings when the dimension of  $X$  is high (e.g.  $p = 100$ ) and the dimension of  $\theta$  is low (e.g.  $q = 1, 2$ ). Given a prior  $\pi$  for  $\theta$ , our approach is to

- (1) Generate a data set  $\mathcal{D}_\pi = \{(\theta^{(i)}, X^{(i)}), 1 \leq i \leq N\}$  by repeatedly drawing  $\theta^{(i)}$  from  $\pi$  and drawing  $X^{(i)}$  from  $\mathcal{M}$  with  $\theta^{(i)}$ .
- (2) Use  $\mathcal{D}_\pi$  to train a DNN with  $\{X^{(i)}, 1 \leq i \leq N\}$  as input and  $\{\theta^{(i)}, 1 \leq i \leq N\}$  as target,
- (3) Run ABC Algorithm 2 with prior  $\pi$  and the DNN estimator  $\hat{\theta}(X)$  as summary statistic.

Our motivation for training a DNN to construct a summary statistic is that the resulting statistic should approximate the posterior mean  $S(X) = \hat{\theta}(X) \approx \mathbb{E}_\pi(\theta|X)$ .

### 2.1. Posterior Means as Summary Statistics

The main advantage of using the posterior mean  $\mathbb{E}_\pi(\theta|X)$  as a summary statistic is that the ABC posterior distribution  $\pi_{ABC}^\epsilon(\theta)$  will then have the same mean as exact posterior distribution in the limit of  $\epsilon \rightarrow 0$ . That is to say,  $\mathbb{E}_\pi(\theta|X)$  does not lose any first-order information when summarizing  $X$ . As Fearnhead and Prangle (2012) discussed, using the posterior mean of the parameter itself as the summary statistic maximizes point-estimation accuracy under squared-error loss. We here provide a simple derivation for this extension in Theorem 1 based on conditioning, which is different to Fearnhead and Prangle (2012)'s proof via density-based calculation. We also provide in supplementary material an extension of Theorem 1 and show the convergence of the posterior expectation of  $b(\theta)$  under the posterior obtained by ABC using  $S_b(X) = \mathbb{E}_\pi[b(\theta)|X]$  as the

summary statistic. Such an extension further establishes a global approximation result of posterior distribution.

**Theorem 1.** *Assume  $\mathbb{E}_\pi|\theta| < \infty$ , then statistic  $S(x) = \mathbb{E}_\pi(\theta|X = x)$  is well defined. ABC procedure with observed data  $x_{obs}$ , summary statistics  $S$ , norm  $\|\cdot\|$  and tolerance threshold  $\epsilon$  produces a posterior distribution  $\pi_{ABC}^\epsilon$ . Then we have*

$$\|\mathbb{E}_{\pi_{ABC}^\epsilon}\theta - S(x_{obs})\| < \epsilon,$$

and

$$\lim_{\epsilon \rightarrow 0} \mathbb{E}_{\pi_{ABC}^\epsilon}\theta = \mathbb{E}_\pi(\theta|X = x_{obs}).$$

*Proof.* First, we show  $S(X) = \mathbb{E}_\pi(\theta|X)$  is a version of conditional expectation of  $\theta$  given  $S(X)$ , i.e.  $S(X) = \mathbb{E}_\pi(\theta|S(X))$ . Denote by  $\sigma(X), \sigma(S(X))$  the  $\sigma$ -algebras of  $X$  and  $S(X)$ , respectively. Clearly  $S(X) = \mathbb{E}_\pi(\theta|X)$  is measurable with respect to  $\sigma(S(X))$ . For any event  $A \in \sigma(S(X))$ ,

$$\begin{aligned} \mathbb{E}_\pi[S(X)\mathbb{I}_A] &= \mathbb{E}_\pi[\mathbb{E}_\pi(\theta|X)\mathbb{I}_A] && \text{(by definition of } S) \\ &= \mathbb{E}_\pi\{\mathbb{E}_\pi[\mathbb{E}_\pi(\theta|X)\mathbb{I}_A|S(X)]\} && \text{(by tower property)} \\ &= \mathbb{E}_\pi\{\mathbb{E}_\pi[\mathbb{E}_\pi(\theta|X)|S(X)]\mathbb{I}_A\} && \text{(since } A \in \sigma(S(X))) \\ &= \mathbb{E}_\pi\{\mathbb{E}_\pi(\theta|S(X))\mathbb{I}_A\} && \text{(by tower property and } \sigma(S(X)) \subseteq \sigma(X)) \end{aligned}$$

Next, write

$$\begin{aligned} \mathbb{E}_{\pi_{ABC}^\epsilon}b(\theta) &= \mathbb{E}_\pi[b(\theta)|\|S_b(X) - S_b(x_{obs})\| < \epsilon] \\ &= \mathbb{E}_\pi[\mathbb{E}_\pi[b(\theta)|S_b(X)]|\|S_b(X) - S_b(x_{obs})\| < \epsilon] \\ &= \mathbb{E}_\pi[S_b(X)|\|S_b(X) - S_b(x_{obs})\| < \epsilon] \end{aligned}$$

which, due to Jensen's inequality, implies that

$$\begin{aligned} \|\mathbb{E}_{\pi_{ABC}^\epsilon}b(\theta) - S_b(x_{obs})\| &= \|\mathbb{E}_\pi[S_b(X)|\|S_b(X) - S_b(x_{obs})\| < \epsilon] - S_b(x_{obs})\| \\ &\leq \mathbb{E}_\pi[\|S_b(X) - S_b(x_{obs})\| | \|S_b(X) - S_b(x_{obs})\| < \epsilon] \\ &< \epsilon \end{aligned}$$

Letting  $\epsilon \rightarrow 0$  yields  $\mathbb{E}_{\pi_{ABC}^\epsilon}b(\theta) \rightarrow S_b(x_{obs}) = \mathbb{E}_\pi[b(\theta)|X = x_{obs}]$ .  $\square$

However, users of Bayesian inference generally desire more than just point estimates: ideally, one approximates the posterior  $\pi(\theta|x_{obs})$  globally. We observe



that such a global approximation result is possible: if one considers a basis of functions on the parameters,  $b_1(\theta), \dots, b_K(\theta)$ , and uses  $(\mathbb{E}_\pi(b_1(\theta)|X), \dots, \mathbb{E}_\pi(b_K(\theta)|X))$  as the summary statistic, the ABC posterior distribution weakly converges to the exact posterior distribution as  $\epsilon \rightarrow 0$  and  $K \rightarrow \infty$  at the appropriate rate. We state our approximation theorem in the supplementary material.

It is also worth noting that there is a nice connection between the posterior mean and the sufficient statistics, especially minimal sufficient statistics in the exponential family. Suppose there exists sufficient statistic  $S^*$  for  $\theta$ , then  $S(X) = \mathbb{E}_\pi(\theta|X)$  is a function of  $S^*$ . In the special case of exponential family with minimal sufficient statistic  $S^*$  and parameter  $\theta$ , the posterior mean  $S(X) = \mathbb{E}_\pi(\theta|X)$  is a one-to-one function of  $S^*(X)$ , and thus is a minimal sufficient statistic.

## 2.2. Structure of Deep Neural Network

At a high level, a deep neural network represents a function for transforming input vector  $X$  into output  $\hat{\theta}(X)$ . The structure of a neural network can be described as a series of  $L$  nonlinear transformations applied to  $X$ . Each of these  $L + 1$  transformations is described as a *layer*: where the original input is  $X$ , the output of the first transformation is the 1st layer, the output of the second transformation is the 2nd layer, and so on, with the output as the  $(L + 1)$ th layer. The layers 1 to  $L$  are called *hidden layers* because they represent intermediate computations, and we let  $H^{(l)}$  denote the  $l$ th hidden layer. Then the explicit form of the network is

$$\begin{aligned} H^{(1)} &= \tanh \left( W^{(0)} H^{(0)} + b^{(0)} \right), \\ H^{(2)} &= \tanh \left( W^{(1)} H^{(1)} + b^{(1)} \right), \\ &\dots \\ H^{(L)} &= \tanh \left( W^{(L-1)} H^{(L-1)} + b^{(L-1)} \right), \\ \hat{\theta} &= W^{(L)} H^{(L)} + b^{(L)}. \end{aligned}$$

where  $H^{(0)} = X$  is the input,  $\hat{\theta}$  is the output,  $W^{(l)}$  and  $b^{(l)}$  are the parameters controlling how the inputs of layer  $l$  are transformed into the outputs of layer  $l$ . Let  $n^{(l)}$  denote the size of the  $l$ th layer: then  $W^{(l)}$  is an  $n^{(l+1)} \times n^{(l)}$  matrix, called the *weight matrix*, and  $b^{(l)}$  is an  $n^{(l+1)}$ -dimensional vector, called the *bias vector*. The  $n^{(l)}$  components of each layer  $H^{(l)}$  are also described evocatively as

“neurons” or “hidden units”. Figure 1 illustrates an example of 2-layer DNN with 5 neurons in the 1st hidden layer and 3 neurons in the 2nd hidden layer, for input data  $X \in \mathbb{R}^4$ .

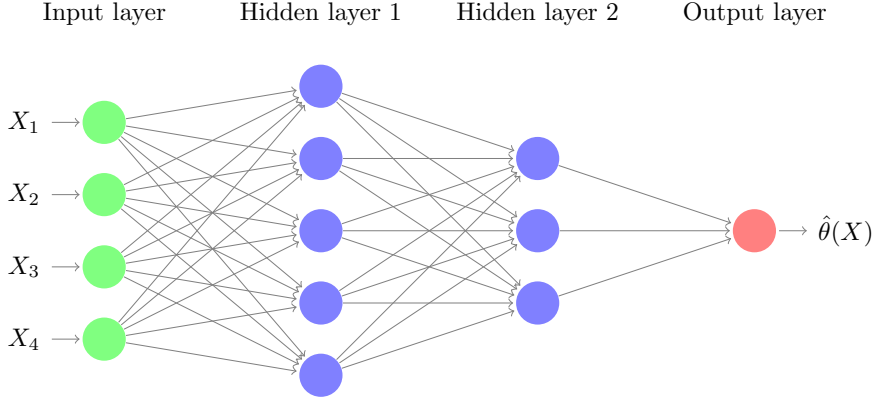
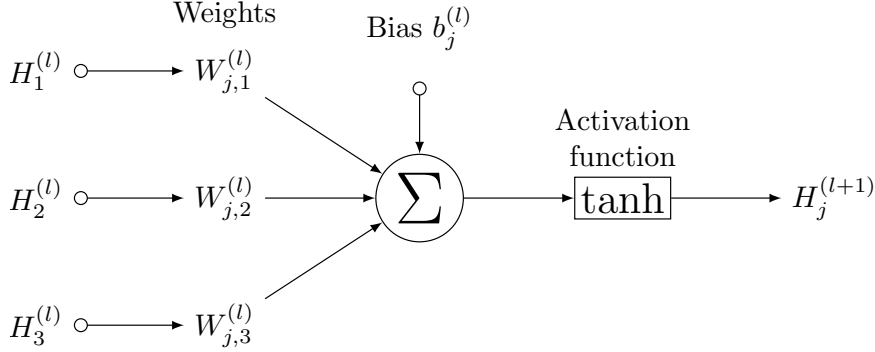


Figure 1: An example of two-layer DNN

The role of layer  $l+1$  is to apply a nonlinear transformation to the outputs of layer  $l$ ,  $H^{(l)}$ , and then output the transformed outputs as  $H^{(l+1)}$ . First, a linear transformation is applied to the previous layer  $H^{(l)}$ , yielding  $W^{(l)}H^{(l)} + b^{(l)}$ . The nonlinearity (in this case tanh) is applied to each element of  $W^{(l)}H^{(l)} + b^{(l)}$  to yield the output of the current layer,  $H^{(l+1)}$ . The nonlinearity is traditionally called the “activation” function, drawing an analogy to the properties of biological neurons. We choose the function tanh as an activation function due to smoothness and computational convenience. Other popular choices for activation function are sigmoid( $t$ ) =  $\frac{1}{1+\exp(-t)}$  and ReLU( $t$ ) =  $\max\{t, 0\}$ . To better explain the activity of each individual neuron, we illustrate how neuron  $j$  in the hidden layer  $l+1$  works in Figure 2.

The output layer takes the top hidden layer  $H^{(L)}$  as input and predicts  $\hat{\theta} = W^{(L)}H^{(L)} + b^{(L)}$ . Note that in many existing applications of deep learning (e. g. computer vision and natural language processing), the goal is to predict a categorical target. In those cases, it is common to use a softmax transformation in the output layer. However, since our goal is prediction rather than classification, it suffices to use a linear transformation.

### 2.3. Approximate Posterior Mean by DNN

Figure 2: Neuron  $j$  in the hidden layer  $l + 1$ 

We use the DNN to construct a summary statistic: a function which maps  $x$  to an approximation of  $\mathbb{E}_\pi(\theta|X)$ . First, we generate a training set  $\mathcal{D}_\pi = \{(\theta^{(i)}, X^{(i)}), 1 \leq i \leq N\}$  by drawing samples from the joint distribution  $\pi(\theta, x)$ . Next, we train the DNN to minimize the squared error loss between training target  $\theta^{(i)}$  and estimation  $\hat{\theta}(X^{(i)})$ . In order words, we minimize an objective function of the DNN parameters  $W^{(0)}, b^{(0)}, \dots, W^{(L)}, b^{(L)}$ , which can be written as

$$\frac{1}{N} \sum_{i=1}^N \|\theta^{(i)} - \hat{\theta}(X^{(i)})\|_2^2.$$

We optimize the objective function by using stochastic gradient descent, computing the derivatives using backpropagation (LeCun, Bottou, Bengio and Haffner (1998)). See the supplementary material for details.

Our approach is based on the fact that any function which minimizes the squared-error risk for predicting  $\theta$  from  $x$  may be viewed as an approximation of the posterior mean  $\mathbb{E}_\pi(\theta|x)$ , since the posterior mean  $\mathbb{E}_\pi(\theta|X)$  is the minimizer of the squared-error risk  $\mathbb{E}_\pi\|\theta - \hat{\theta}(X)\|_2^2$ . Hence, any number of supervised learning approaches could be used to construct a prediction rule for predicting  $\theta$  from  $x$ , and thereby provide an approximation of  $\mathbb{E}_\pi(\theta|x)$ . Since in many applications of ABC, we can expect  $\mathbb{E}_\pi(\theta|x)$  to be a highly nonlinear and smooth function, it is important to choose a supervised learning approach which has the power to approximate such nonlinear smooth functions.

Therefore, deep neural networks appear to be a good choice given their rich

representational power for approximating nonlinear functions. Indeed, it is speculated that by increasing the depth and width of the network, the DNN gains the power to approximate any continuous function; however, rigorous proof of the approximation properties of DNNs remains an important open problem (Faragó and Lugosi (1993); Sutskever and Hinton (2008); Le Roux and Bengio (2010)). In any case, in order to take advantage of the representational power (and avoid overfitting), it is important to have a sufficiently large training set. Fortunately, in applications of Approximate Bayesian Computation, an arbitrarily large training set can be generated. Furthermore, it is worth noting that both dataset generation and training can be parallelized.

### 3. Example: Ising Model

#### 3.1. Model Description and Experiment Design

The Ising model consists of discrete variables ( $+1$  or  $-1$ ) arranged in a lattice (Figure 3).

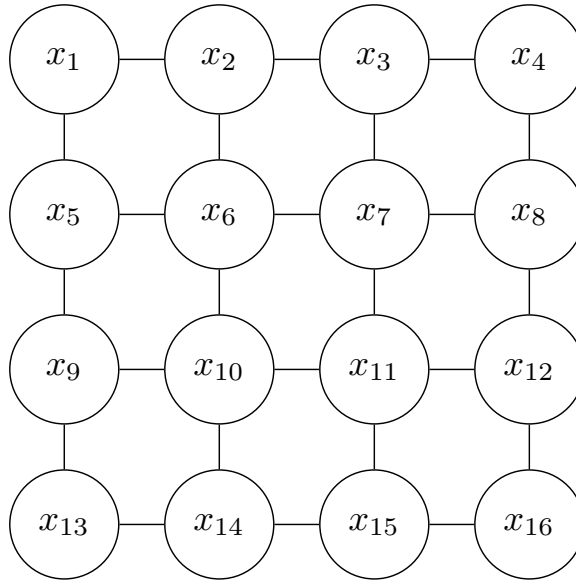


Figure 3: Ising model on  $4 \times 4$  lattice

Each binary variable, called a spin, is allowed to interact with its neighbors. The inverse-temperature parameter  $\theta > 0$  characterizes the extent of interaction.

Given  $\theta$ , the probability mass function of Ising model on  $m \times m$  lattice is

$$p(X|\theta) = \frac{\exp\left(\theta \sum_{j \sim k} X_j X_k\right)}{Z_\theta}$$

where  $X_j \in \{-1, +1\}$ ,  $j \sim k$  means  $X_j$  and  $X_k$  are neighbors, and the normalizing constant

$$Z_\theta = \sum_{x' \in \{-1, +1\}^{m \times m}} \exp\left(\theta \sum_{j \sim k} x'_j x'_k\right).$$

Since the normalizing constant requires an exponential-time computation, the probability mass function  $p(x|\theta)$  is intractable except in small cases. The Ising model is a natural exponential family with minimal sufficient statistics

$$S^*(X) = \sum_{j \sim k} X_j X_k,$$

which is a highly non-linear function in high-dimensional space  $\{-1, +1\}^{10 \times 10}$ .

Despite of the unavailability of probability mass function, data  $X$  can be still simulated given  $\theta$  using Monte Carlo methods such as Metropolis algorithm (Asmussen and Glynn (2007)). The Ising model on a square lattice undergoes a phase transition as  $\theta$  increases. When  $\theta$  is small, the spins are disordered. When  $\theta$  is large enough, the spins tend to have the same sign due to the strong neighbour-to-neighbour interactions (Onsager (1944)). The phase transition of Ising model on infinite lattice has a critical point  $\theta_c = 0.4406$ . Ising model on finite lattice is slightly different: its phase transition smoothly occurs around that critical point (Fearnhead and Prangle (2012)). So we consider Ising model on  $10 \times 10$  lattice and choose the prior  $\pi \sim \text{Exp}(0.4406)$  for  $\theta$ .

The DNN-based summary statistic  $S(X)$  approximates the posterior mean, which in turn is an increasing function of the sufficient statistic  $S^*(X)$ , since the Ising model is an exponential family. Hence, for the evaluation purpose, we compare our DNN-based summary statistic  $S(X)$  to the minimal sufficient statistic  $S^*(X)$ , and further compare the two resulting ABC posterior distributions. It's a challenging task to constructing such a summary statistic  $S(X)$  due to the high non-linearity of the sufficient statistic  $S^*(X) = \sum_{j \sim k} X_j X_k$  in the high-dimensional space  $\{-1, +1\}^{10 \times 10}$  where  $X$  lies in. Given the prior  $\pi \sim \text{Exp}(0.4406)$  for  $\theta$ , we generate a training set by Metropolis algorithm, and

train a DNN to learn summary statistic  $S$ . We then compare  $S(X)$  to the known sufficient statistic  $S^*(X)$ . Figure 4 outlines this experimental scheme.

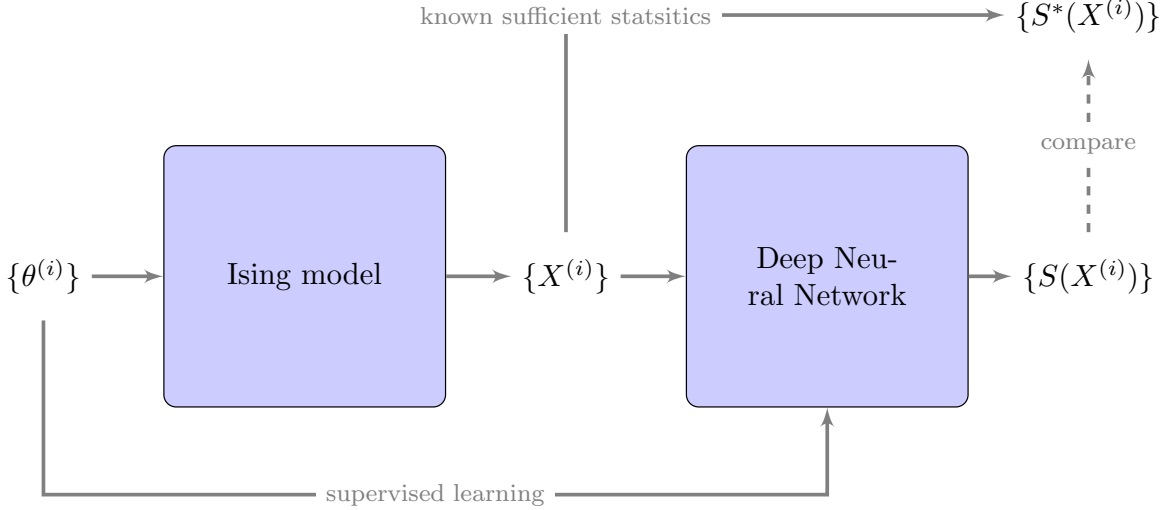


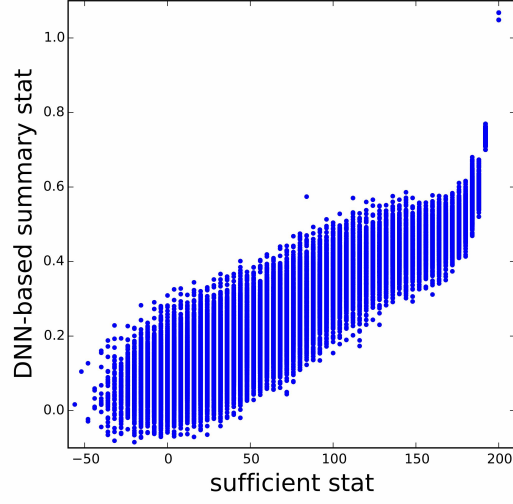
Figure 4: Experimental design on Ising model

### 3.2. Results

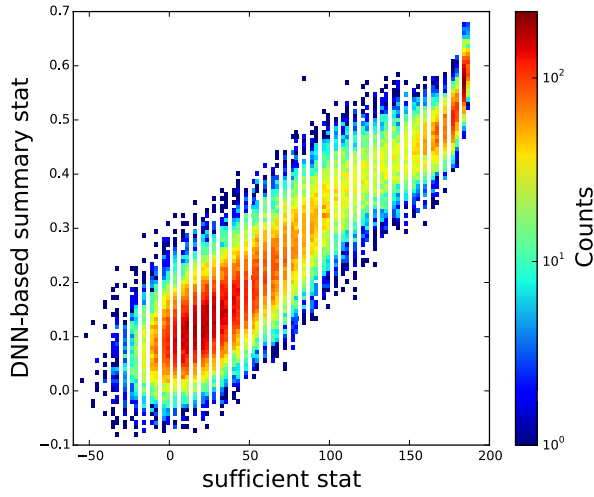
Given the prior  $\pi \sim \text{Exp}(0.4406)$ , we generate a training set of size  $10^6$  and a testing set of size  $10^5$  from the Ising model on  $10 \times 10$  lattice. As discussed in Section 3.1, large  $\theta$  tends to produce  $X$  which consists entirely of either  $+1$  or  $-1$ . Consistent with this, 22.8% of generated test instances have spins with the same sign, which results in sufficient statistic  $S^* = 200$ , and 4.7% of the test instances have all but one spins with the same sign, resulting in  $S^* = 192$ .

A three-layer DNN with  $n^{(1)} = 500$ ,  $n^{(2)} = 200$ , and  $n^{(3)} = 100$  in each hidden layer is trained to predict  $\theta$  from  $x$ ; we define the summary statistic  $S(x)$  as the output of the DNN. Figure 5a displays a scatterplot which compares the DNN-based statistic  $S$  and sufficient statistic  $S^*$ . Each point in the scatterplot represents to  $(S^*(x), S(x))$  for a single instance  $x$  in the testing set. A large number of the instances are concentrated at  $S^* = 200$  and  $S^* = 192$ , which appear as points in the top-right corner of the scatterplot. These two cases are relatively uninteresting, so in Figure 5b we display a heatmap of  $(S(x), S^*(x))$  excluding the instances with  $S^* = 192, 200$ . It shows that the DNN-based summary statistic

approximates a strictly increasing function of sufficient statistic.

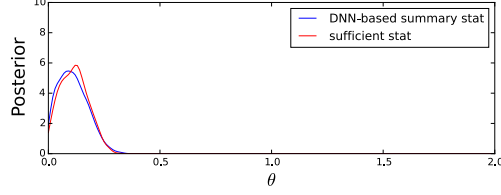
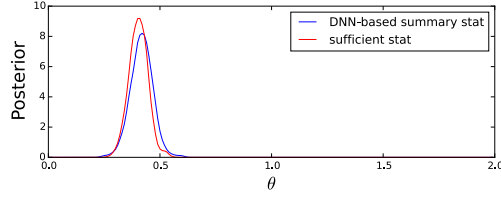
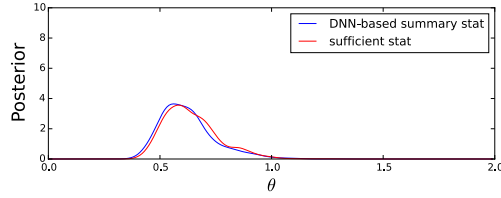
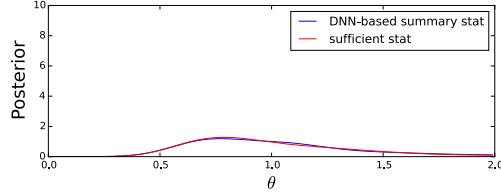


(a) Scatterplot of  $10^5$  test instances. Each point in the scatterplot represents to  $(S^*(x), S(x))$  for a single test instance  $x$ .



(b) Heatmap excluding instances with sufficient statistic  $S^* = 192, 200$

Figure 5: DNN-based summary statistic  $S$  v.s. sufficient statistic  $S^*$  on the test dataset.

(a) True  $\theta = 0.2$ (b) True  $\theta = 0.4$ (c) True  $\theta = 0.6$ (d) True  $\theta = 0.8$ Figure 6: ABC posterior distributions for  $x_{obs}$  generated with true  $\theta = 0.2, 0.4, 0.6, 0.8$ .

Next, two ABC posterior distributions are obtained with  $S^*$  and  $S$  as summary statistic, respectively. For the sufficient statistic  $S^*$ , we set the tolerance level  $\epsilon = 0$  so that the ABC posterior sample follows the exact posterior distribution  $\pi(\theta|X = x_{obs})$ . For the DNN-based summary statistic  $S$ , we set the tolerance threshold  $\epsilon$  small enough so that 0.1% of  $10^6$  proposed  $\theta$ 's are accepted. We repeat the comparison for 4 different observed data  $x_{obs}$ , which are generated from  $\theta = 0.2, 0.4, 0.6, 0.8$ , respectively; in each case, we compare the posterior



obtained from  $S^*$  with the posterior obtained from  $S$  in Figure 6.

It is also worth highlighting the case with true  $\theta = 0.8$  in Figure 6d. Since with high probability the spins  $X_i$  have the same sign when  $\theta$  is large, it becomes difficult to distinguish different values of  $\theta$  above the critical point  $\theta_c$  based on the data  $x_{obs}$ . Hence we should expect the posterior be small below  $\theta_c$  and have a similar shape to the prior distribution above  $\theta_c$ . Both of the ABC posteriors demonstrate this property.

#### 4. Example: Moving-average Model

##### 4.1. Model Description and Experiment Design

The moving-average model is widely used in time series analysis. Each value in the time series is described by a linear combination of current and previous unobserved white noise error terms. Let  $X_1, \dots, X_p$  denote the observations in time series. Then the moving-average model of order  $q$ , denoted by  $MA(q)$  is given by

$$X_j = Z_j + \theta_1 Z_{j-1} + \theta_2 Z_{j-2} + \dots + \theta_q Z_{j-q}, \quad j = 1, \dots, p$$

where  $Z_j$  are unobserved white noise error terms. In our experiments, we let  $Z_j \stackrel{i.i.d.}{\sim} N(0, 1)$  in order to enable exact calculation of the posterior distribution  $\pi(\theta|x_{obs})$ , so that we can evaluate the accuracy of the ABC posterior distribution. In the case that  $Z_j$ 's are non-Gaussian (e.g. Student's t-distribution), the exact posterior distribution  $\pi(\theta|x_{obs})$  becomes intractable to compute, but ABC is still applicable.

Approximate Bayesian Computation has been applied to study the posterior distribution of  $MA(2)$  model using the auto-covariance as the summary statistic Marin, Pudlo, Robert and Ryder (2012). The auto-covariance is a natural choice for the summary statistic in the  $MA(2)$  model because it converges to a one-to-one function of underlying parameter  $\theta = (\theta_1, \theta_2)$  in probability as  $p \rightarrow \infty$ , by the weak law of large number

$$AC_1 = \frac{1}{p-1} \sum_{j=1}^{p-1} X_j X_{j+1} \xrightarrow{\mathbb{P}} \mathbb{E}(X_1 X_2) = \theta_1 + \theta_1 \theta_2$$

$$AC_2 = \frac{1}{p-2} \sum_{j=1}^{p-2} X_j X_{j+2} \xrightarrow{\mathbb{P}} \mathbb{E}(X_1 X_3) = \theta_2.$$

Since the MA(2) model is identifiable over the triangular region

$$\theta_1 \in [-2, 2], \quad \theta_2 \in [-1, 1], \quad \theta_2 \pm \theta_1 \geq -1,$$

we consider a uniform prior  $\pi$  over this region, and proceed to construct a summary statistic similarly to the previous example. As before, we use the prior  $\pi$  to generate a training set and train a three-layer DNN to predict  $\theta$  based on  $X$ . The two-dimensional estimator  $(S_1, S_2)$  implicitly defined by DNN is taken as the summary statistic.

We perform a number of experiments to compare the auto-covariance, the DNN-based summary statistic and semi-automatic summary statistic. In each experiment, we generate some true parameter  $\theta$  from the prior, and draw the observed data  $x_{obs}$ . The exact posterior distribution given  $x_{obs}$  is numerically computed. Then we compute ABC posterior distributions using the auto-covariance statistic  $(AC_1, AC_2)$ , the DNN-based summary statistics  $(S_1, S_2)$ , and the semi-automatic summary statistic, respectively. The three resulting approximate posterior distributions are compared to the exact posterior distribution and evaluated in terms of the accuracies of posterior mean of  $\theta$ , posterior marginal variances of  $\theta_1, \theta_2$ , and the posterior correlation between  $(\theta_1, \theta_2)$ .

#### 4.2. Results

Given the prior  $\pi$ , we generate a training set of size  $10^6$  and a testing set of size  $10^5$  from the MA(2) model, where each instance is a time series of length  $p = 100$ . Then a three-layer DNN with  $n^{(1)} = 500$ ,  $n^{(2)} = 200$ , and  $n^{(3)} = 100$  is trained to predict  $\theta$  from  $x$ . The summary statistic  $S(x)$  is defined as the output of the DNN. As shown in Figure 7, the DNN predicts  $\theta_1, \theta_2$  in the test set with mean squared errors (MSEs) of 0.021 and 0.024, respectively. In comparison, the semi-automatic method achieves MSEs of 0.678 and 0.189.

For observed data  $x_{obs}$  which is generated by true parameter  $\theta$  drawn from  $\pi$ , we run ABC procedures with three different choices of summary statistic: the DNN-based summary statistic, the auto-covariance, and also the semi-automatic summary statistic. The tolerance threshold  $\epsilon$  is set to accept 0.1% of  $10^5$  proposed data points in ABC procedures. Figure 8 compares the true posterior with the posterior draws obtained by ABC, for a particular  $x_{obs}$  with  $\theta = (0.6, 0.2)$ .

Using the DNN-based summary statistic results in a more accurate ABC posterior distribution than either the ABC posterior distribution obtained by

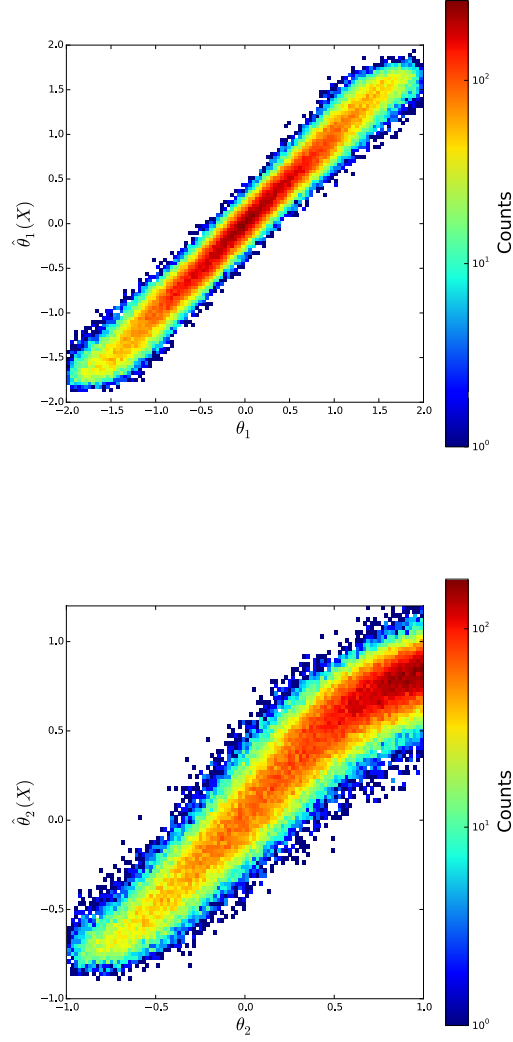


Figure 7: DNN predicting the parameters of MA(2) model for  $10^5$  test instances

using the auto-covariance statistic or the semi-automatic construction. One of the important features of the DNN-based summary statistic is its resulting ABC posterior correctly captures the correlation between  $\theta_1$  and  $\theta_2$ , while the auto-covariance statistic and the semi-automatic statistic appears to be insensitive to

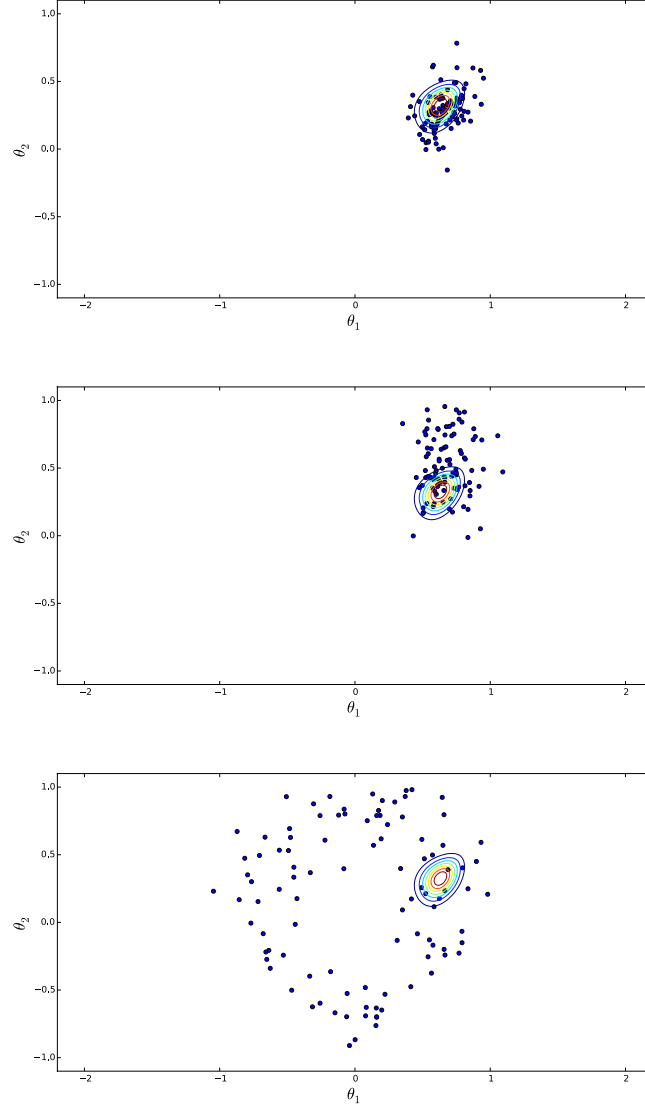


Figure 8: ABC posterior distributions (top: DNN-based summary statistics, middle: auto-covariance, bottom: semi-automatic construction) for observed data  $x_{obs}$  generated with  $\theta = (0.6, 0.2)$ , compared to the exact posterior distribution contours.

this information (Table 1).

We further repeated the comparison for 100 different  $x_{obs}$ . As Table 2 shows, the ABC procedure with the DNN-based statistic better approximates the pos-

Table 1: Mean and covariance of exact/ABC posterior distributions for observed data  $x_{obs}$  generated with  $\theta = (0.6, 0.2)$

Posterior	Exact	ABC (DNN)	ABC (auto-cov)	ABC (semi-auto)
mean( $\theta_1$ )	0.6243	0.6567	0.6760	-0.0030
mean( $\theta_2$ )	0.3190	0.2810	0.5113	0.2142
std( $\theta_1$ )	0.0910	0.1174	0.1386	0.5540
std( $\theta_2$ )	0.1091	0.1504	0.2230	0.4169
cor( $\theta_1, \theta_2$ )	0.3664	0.4110	0.0471	0.0205

terior moments than the ABC posteriors using the auto-covariance statistic and semi-automatic construction.

Table 2: Mean squared error between mean and covariance of exact/ABC posterior distributions over 100 datasets

	ABC (DNN)	ABC (auto-cov)	ABC (semi-auto)
MSE for mean( $\theta_1$ )	0.0100	0.0111	0.5277
MSE for mean( $\theta_2$ )	0.0119	0.0184	0.1466
MSE for std( $\theta_1$ )	0.0040	0.0041	0.4603
MSE for std( $\theta_2$ )	0.0042	0.0065	0.1002
MSE for cor( $\theta_1, \theta_2$ )	0.0728	0.1886	0.2976

## 5. Discussion

The problem that we address in this article is how to automatically construct low-dimensional and informative summary statistics for ABC methods, with minimal need for expert knowledge. We base our approach on theoretical results on the desirable properties of the posterior mean as a summary statistic for ABC. However, since the posterior mean is generally intractable, we take advantage of the representational power of DNNs to construct an approximation of the posterior mean as a summary statistic. In contrast to many existing methods that select or construct summary statistics from ad-hoc candidate summary statistics, our approach automatically searches through a rich class of nonlinear transformations of the input data to yield an appropriate summary statistic.

Although we can only heuristically justify our choice of DNNs to construct the approximation (due to a lack of rigorous theory on the approximation properties of DNNs), we obtain promising empirical results. In two examples, the Ising model and the moving-average model, we find that the the DNN-based statistics are good approximations of the posterior means, and further result in high-quality approximations to the true posterior distribution.

## Supplementary Materials

We first present in supplementary material an extension of Theorem 1 and show the convergence of the posterior expectation of  $b(\theta)$  under the posterior obtained by ABC using  $S_b(X) = \mathbb{E}_\pi [b(\theta)|X]$  as the summary statistic. Such an extension further establishes a global approximation result of posterior distribution as Theorem 2. Implementation details of backpropagation and stochastic gradient descent algorithms when training deep neural network are provided. The derivatives of squared error loss function with respect to network parameters are computed. They are used by stochastic gradient descent algorithms to train deep neural networks.

## Acknowledgements

The authors gratefully acknowledge National Science Foundation grants DMS1407557 and DMS1330132.

## References

- Asmussen, S., and Glynn, P. W. (2007). *Stochastic simulation: Algorithms and analysis* (Vol. 57). Springer Science & Business Media.
- Beaumont, M. A., Zhang, W., and Balding, D. J. (2002). Approximate Bayesian computation in population genetics. *Genetics*, **162**(4), 2025-2035.
- Lopes, J. S. and Beaumont, M. A. (2010). ABC: a useful Bayesian tool for the analysis of population data. *Infection, Genetics and Evolution*, **10**(6), 825-832.
- Beaumont, M. A. (2010). Approximate Bayesian computation in evolution and ecology. *Annual review of ecology, evolution, and systematics*, **41**, 379-406.
- Csilléry, K., Blum, M. G., Gaggiotti, O. E. and François, O. (2010). Approximate Bayesian computation (ABC) in practice. *Trends in ecology & evolution*, **25**(7), 410-418.

- Marin, J. M., Pudlo, P., Robert, C. P. and Ryder, R. J. (2012). Approximate Bayesian computational methods. *Statistics and Computing*, **22**(6), 1167-1180.
- Toni, T., Welch, D., Strelkowa, N., Ipsen, A. and Stumpf, M. P. (2009). Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. *Journal of the Royal Society Interface*, **6**(31), 187-202.
- Sunnåker, M., Busetto, A. G., Numminen, E., Corander, J., Foll, M. and Dessimoz, C. (2013). Approximate Bayesian computation. *PLoS Comput. Biol.*, **9**(1), e1002803.
- Tavaré, S., Balding, D. J., Griffiths, R. C. and Donnelly, P. (1997). Inferring coalescence times from DNA sequence data. *Genetics*, **145**(2), 505-518.
- Fu, Y. X. and Li, W. H. (1997). Estimating the age of the common ancestor of a sample of DNA sequences. *Molecular biology and evolution*, **14**(2), 195-199.
- Weiss, G. and von Haeseler, A. (1998). Inference of population history using a likelihood approach. *Genetics*, **149**(3), 1539-1546.
- Pritchard, J. K., Seielstad, M. T., Perez-Lezaun, A., and Feldman, M. W. (1999). Population growth of human Y chromosomes: a study of Y chromosome microsatellites. *Molecular biology and evolution*, **16**(12), 1791-1798.
- Blum, M. G., Nunes, M. A., Prangle, D., Sisson, S. A. and others (2013). A comparative review of dimension reduction methods in approximate Bayesian computation. *Statistical Science*, **28**(2), 189-208.
- Lehmann, E. L. and Casella, G. (1998). *Theory of point estimation* (Vol. 31). Springer Science & Business Media.
- Marjoram, P., Molitor, J., Plagnol, V., and Tavaré, S. (2003). Markov chain Monte Carlo without likelihoods. *Proceedings of the National Academy of Sciences*, **100**(26), 15324-15328.
- Sisson, S. A., Fan, Y. and Tanaka, M. M. (2007). Sequential monte carlo without likelihoods. *Proceedings of the National Academy of Sciences*, **104**(6), 1760-1765.
- Joyce, P. and Marjoram, P. (2008). Approximately sufficient statistics and Bayesian computation. *Statistical applications in genetics and molecular biology*, **7**(1).
- Nunes, M. A. and Balding, D. J. (2010). On optimal selection of summary statistics for approximate Bayesian computation. *Statistical applications in genetics and molecular biology*, **9**(1).
- Boulesteix, A. L. and Strimmer, K. (2007). Partial least squares: a versatile tool for the analysis of high-dimensional genomic data. *Briefings in bioinformatics*, **8**(1), 32-44.

- Wegmann, D., Leuenberger, C. and Excoffier, L. (2009). Efficient approximate Bayesian computation coupled with Markov chain Monte Carlo without likelihood. *Genetics*, **182**(4), 1207-1218.
- Blum, M. G. and François, O. (2010). Non-linear regression models for Approximate Bayesian Computation. *Statistics and Computing*, **20**(1), 63-73.
- Fearnhead, P. and Prangle, D. (2012). Constructing summary statistics for approximate Bayesian computation: semi-automatic approximate Bayesian computation. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **74**(3), 419-474.
- Hinton, G. E. and Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science*, **313**(5786), 504-507.
- Hinton, G. E., Osindero, S., and Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural computation*, **18**(7), 1527-1554.
- Bengio, Y., Courville, A., and Vincent, P. (2013). Representation learning: A review and new perspectives. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **35**(8), 1798-1828.
- Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, **61**, 85-117.
- LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, **86**(11), 2278-2324.
- Faragó, A., and Lugosi, G. (1993). Strong universal consistency of neural network classifiers. *Information Theory, IEEE Transactions on*, **39**(4), 1146-1151.
- Sutskever, I., and Hinton, G. E. (2008). Deep, narrow sigmoid belief networks are universal approximators. *Neural Computation*, **20**(11), 2629-2636.
- Le Roux, N., and Bengio, Y. (2010). Deep belief networks are compact universal approximators. *Neural computation*, **22**(8), 2192-2207.
- Onsager, L. (1944). Crystal statistics. I. A two-dimensional model with an order-disorder transition. *Physical Review*, **65**(3-4), 117.
- Landau, D. P. (1976). Finite-size behavior of the Ising square lattice. *Physical Review B*, **13**(7), 2997.